REMARKS

Claims 1-28 are pending and at issue. Claims 1- 26 and 28 stand rejected under 35 U.S.C. § 102 as anticipated by Flinn I entitled "Energy-aware adaptation for mobile applications." Claim 27 has been confirmed as reciting allowable subject matter and would be allowed if rewritten in independent form including all of the limitations of its base claim and intervening claims.

Flinn I describes a technique in which applications can dynamically modify their behavior to conserve energy through an operating system that monitors energy supply and demand. Flinn I uses a tool called PowerScope, also developed by the authors, to build the energy profile used by these applications. The PowerScope tool is discussed in Flinn I but only briefly, in section 2.1 and in FIG. 1. The official action cites to this discussion of PowerScope in rejecting independent claims 1 and 10. The discussion is incomplete, however, in that Flinn I does not describe how PowerScope performs its data collection or how it creates an energy profile of code. In response to these questions, Applicant's representative obtained a copy of the publication entitled "PowerScope: a tool for profiling the energy usage of mobile applications," written by Flinn and Satyanarayanan ("Flinn II") and which describes the PowerScope application of Flinn I. Flinn I specifically cites to Flinn II at footnote at [8].

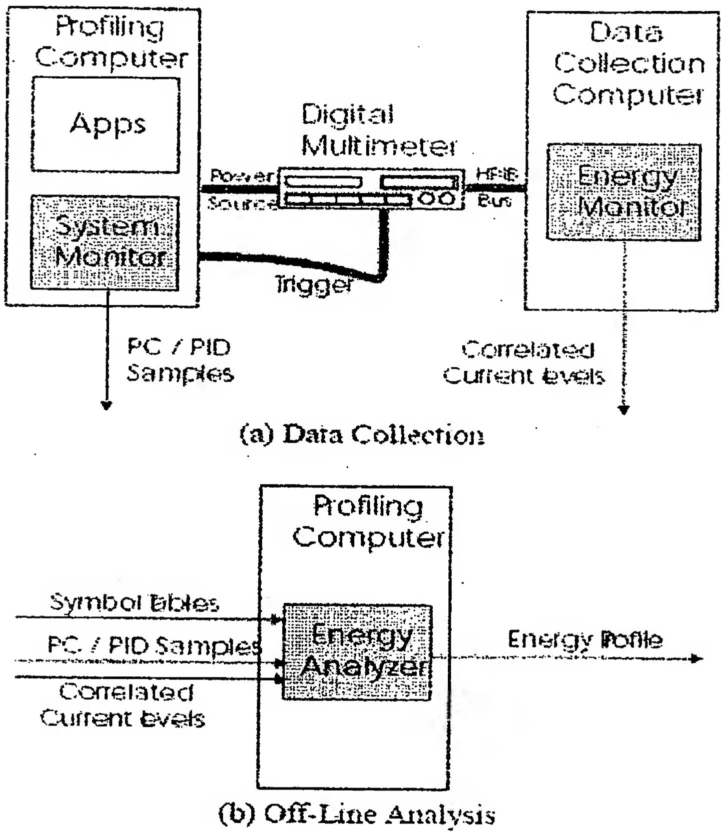
The Flinn II paper is important in that it shows that the energy monitoring tool used in Flinn I is a tool that is <u>time-triggered</u>, not quantum-of-power triggered. Instead of sampling a computer state each time a repeating quantum of power is used on a system, however long it takes for that quantum to be used, Flinn I triggers its sampling each clock cycle of a digital multimeter. That is, while Flinn I does have an energy monitor that stores measured current and a system monitor that monitors a program counter and process identifier, both monitors trigger on a time-based event, the clock signal.

Before discussing Flinn II further, Applicants would like to draw the Examiner's attention to the discussions in the background of the application where various known techniques for monitoring energy usage are discussed, including techniques like that of Flinn II which measure energy usage in response to time-based triggers. Indeed,

deficiencies with time-based triggers, in part, precipitated the need for triggering based on other measures, such as quantums of power used. As the application describes prior art time-based triggering systems:

> [0005] To obtain data about application performance, the tools typically profile application code and quantify the usage of various system resources. Application code is loaded and executed in an environment that is able to monitor and record various performance characteristics during code execution. Monitoring a complex software application in its entirety is very expensive and impractical, however. **Thus, for efficiency purposes, performance analysis tools periodically sample the code executing environment to obtain an 'accurate' measure of performance. This sampling is either time-based or event-based.**
>
> [0006] **In a time-based analysis, the performance tool periodically takes a snapshot of the current state of the system after a predetermined time, or number of clock [sic] cycles.** In an event-based analysis, a snapshot is taken every time a certain event occurs within the system, such as a cache miss or branch mis-predict. The sampled performance statistics are used to build a profile of the performance of the application running on the monitored system. For example, to identify code that causes an excessive number of data cache misses, a performance analysis tool can use event based sampling (the event being a data cache miss) to profile application code and determine which code modules are using memory inefficiently. These code modules may then be optimized.
>
> ...
>
> [0008] To analyze power requirements of mobile applications, performance analysis tools must include mechanisms that enable the profiling of application power consumption. However, simply incorporating power measurement into the existing performance analysis framework will not yield an accurate profile of an applications' power consumption. **Neither time- nor event-based sampling is suited for generating profiles of system power usage that provide an accurate and detailed account of power consumption of different code modules.**

The descriptions of Flinn II are very similar to this prior art description. Figures 1a and 1B of Flinn II show the PowerScope architecture, with Figure 1a being the same as Figure 1 of Flinn I.



This figure shows how PowerScope generates an energy profile. As applications execute on the profiling computer, the System Monitor samples system activity and the Energy Monitor samples power consumption. Later, the Energy Analyzer uses this information to generate an energy profile.

Figure 1. PowerScope Architecture

The Flinn system has three components – a System Monitor, an Energy Monitor, and a Digital Multimeter – which are described generally as follows:

> The System Monitor samples system activity on the profiling computer by periodically recording information which includes the program counter (Pc) and process identifier (PID) of the currently executing process. The Energy Monitor runs on the data-collection computer, and is responsible for collecting and storing current samples. Flinn II, p. 2, col. 2.

Flinn II makes clear that the System Monitor and Energy Monitor are triggered by the digital multimeter clock.[1]

> Because data collection is distributed across two monitor processes, it is essential that some synchronization method ensure that **they collect samples closely correlated in time**. Flinn II, p. 2, col. 2 – p. 3, col. 1.
>
> Our current implementation samples system activity **when triggered by the digital multimeter**. Flinn II, p. 3, col. 1.
>
> **Sample collection is driven by the multimeter clock**. Flinn II, p. 3, col. 2.
>
> **Our original design used the clock of the profiling computer to drive sample collection.** Although simpler to implement, that design had the disadvantage of biasing the profile values of activities correlated to the system clock. **Using the multimeter clock also allows us to generate interrupts at a finer granularity than [sic] that allowed by the kernel stat clock routine. The user may specify the sample period as a parameter when the Energy Monitor is started.** Flinn II, p.3, col. 2.

At no time is the System Monitor instructed to measure the program counter or process identification in response to the measurement of a particular quantum of power. The amount of power used is irrelevant to sampling. At each clock cycle, the multimeter measures current and then, regardless of what that measured current value is at each clock cycle, instructs the System Monitor to sample. The multimeter does not determine an energy level from the current or whether that energy level rises to a particular quantum of power

---

[1] Flinn II describes that as the multimeter measures current upon each clock cycle, the multimeter triggers the System Monitor to correspondingly sample the system:

> Synchronization with the System Monitor is provided by connecting the multimeter's external trigger input and output to pins on the parallel port of the profiling computer. Immediately, after the multimeter takes a current sample, it toggles the value of a parallel port pin. This causes a system interrupt on the profiling computer, during which the System Monitor samples systems activity. Upon completion, the System Monitor triggers the next sample by toggling another parallel port pin (unless profiling has been halted by the pscope_stop system call). The multimeter buffers this trigger until the time to take the next sample arrives. Flinn II, p. 3, col. 2.

From the foregoing, it is clear that Flinn I cannot be said to teach determining when a quantum of power has been used on the machine, and in response to usage of the quantum of power on the machine, triggering sampling of state data of the machine, where the state data indicates a state of code executing on the machine, as generally provided for in claims 1 and 10. The rejection of these claims and the claims depending therefrom are respectfully traversed.

As amended claim 20 recites an apparatus comprising:

> a power measurement module capable of measuring power usage in the apparatus and capable of determining when a quantum of power has been used over a period of time and capable of adjusting granularity of the quantum of power; and
>
> a power sampling module coupled to the power measurement module for sampling state data of the apparatus, where sampling is triggered by an indication from the power measurement module when each of a plurality of the quantums of power has been used; and
>
> a power analysis module that analyzes code executing on the apparatus in response to the sampling of the state data to develop a power profile of the code.

The action points to the Energy Monitor of Flinn I as teaching both the power measurement module and the power sampling module. Applicant respectfully disagrees.

First, the Energy Monitor of Flinn I does not sample state data, as is made clear in Flinn II. Program counter and process identification sampling is achieved by the System Monitor, which runs on an entirely different machine than the Energy Monitor. *Second*, the Energy Monitor does not actually sample power. The Energy Monitor instructs the multimeter to sample current of the System Monitor machine at periodic clock intervals. The Energy Monitor merely receives asynchronously transmitted current sample data and "stores them on disk for later analysis." Flinn II, p. 3, col. 2. *Third*, as noted above, neither the Energy Monitor nor the System Monitor are triggered to store or sample data, respectively, in response to an indication that a particular quantum of power has been used by the machine. Both are triggered by the multimeter clock and irrespective of the amount of power that is delivered or consumed between clock cycles. The total power or energy used

over time is not even determined until after the sampling and then by a separate Energy Analyzer.

Furthermore, claim 20 has been amended to track language similar to that of provisionally-allowed claim 27, specifically that the power measurement module is "capable of adjusting granularity of the quantum of power." Flinn I and II do not teach or suggest this subject matter either.

In light of the foregoing, independent claim 20 and the claims depending therefrom are in condition for immediate allowance.

In view of the above amendment, Applicant believes the pending application is in condition for allowance.

Dated: October 12, 2006

Respectfully submitted,

By ____________

Paul B. Stephens
Registration No.: 47,970
MARSHALL, GERSTEIN & BORUN LLP
233 S. Wacker Drive, Suite 6300
Sears Tower
Chicago, Illinois 60606-6357
(312) 474-6300
Attorney for Applicant